REMARKS

Claims 11-19, 25-31, and 36-41 are pending in the application. Claims 11-19, 25-31, and 36-41 stand rejected under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent No. 5,928,325 to Shaughnessy et al. Claims 1-10, 20-24, and 32-35 stand withdrawn as directed to non-elected inventions.

Reconsideration is requested. No new matter is added. The rejections are traversed. Claims 11-14, 16, 19, 27, 29-30, 38, and 40-41 are amended. Claims 15, 17, 25-26, and 36-37 are canceled. Claims 1-10, 20-24, and 32-35 are canceled as being directed to non-elected inventions. Claims 42-45 are added. Claims 11-14, 16, 18-19, 27-31, and 38-45 remain in the case for consideration.

INTERVIEW SUMMARY

On June 14, 2004, Examiner Duong called the undersigned regarding a restriction requirement. The Examiner divided the claims into two groups. Group I included claims 1-10, 20-24, and 32-35. Group II included claims 11-19, 25-31, and 36-41. The undersigned elected the claims of Group II.

REJECTIONS UNDER 35 U.S.C. § 102(e)

Referring to claim 11, the invention is directed toward a network lurking agent operable in a Scalable Infrastructure system, the network lurking agent comprising: an inquirer designed to place an inquiry in a persistent store called a Space, the Space part of the Scalable Infrastructure system; and a lurker designed to retrieve from the Space a response to the inquiry to determine the availability of a user in an environment.

Referring to claim 14, the invention is directed toward a Scalable Infrastructure system designed to support network lurking, the Scalable Infrastructure system comprising: a persistent store called a Space, the Space part of the Scalable Infrastructure system; an environment setting stored in the Space, the environment setting including the availability of a device in an environment; a network receiving agent designed to receive an inquiry about the availability of the device in the environment from the Space; and a network lurking agent designed to place the inquiry in the Space.

Referring to claim 42, the invention is directed toward a method for using a network lurking agent to electronically lurk to an environment in a Scalable Infrastructure system, the method comprising: identifying an environment of interest; and placing an inquiry as to the

availability of a user in the environment of interest in a persistent store called a Space, the Space part of the Scalable Infrastructure system.

Referring to claim 44, the invention is directed toward an apparatus for using a network lurking agent to electronically lurk to an environment in a Scalable Infrastructure system, the apparatus comprising: means for identifying an environment of interest; and means for placing an inquiry as to the availability of a user in the environment of interest in a persistent store called a Space, the Space part of the Scalable Infrastructure system.

In contrast, Shaughnessy teaches a method for dynamically establishing communication of messages to devices. All messages are delivered to a central agent server. The central agent server determines which devices are associated with the message's recipient, and which networks service those devices. The central agent server then polls the networks to determine which devices are available. If any devices are available, the central agent server transforms the message according to rules appropriate for the network(s) servicing the devices, and delivers the transformed message.

The claims have been amended to make specific mention of the concept of the persistent store termed a *Space*. Spaces are exemplified by JavaSpaces. The following are some basic principles about JavaSpaces. For more details, the Examiner is referred to JavaSpaces Principles, Patterns and Practice,
by Eric Freeman, Susanne Hupfer, and Ken Arnold.

- JavaSpaces is the realization of "tuple spaces" from David Gelernter.
- "A space is a shared, network-accessible repository for objects. Processes use the repository as a persistent object storage and exchange mechanism; instead of communicating directly, they coordinate by exchanging objects through spaces. Processes perform simple operations to write new objects into a space, take objects from a space, or read (make a copy of) objects in a space. When taking or reading objects, processes use a simple value-matching lookup to find the objects that matter to them. If a matching object isn't found immediately, then a process can wait until one arrives. Unlike conventional object stores, processes don't modify objects in the space or invoke their methods directly -- while there, objects are just passive data. To modify an object, a process must explicitly remove it, update it, and reinsert it into the space." JavaSpaces Principles, Patterns, and Practice Freeman, Hupfer, Arnold.
- Differences between a table row and a tuple
- Spaces only allow for primitive type matching, exact field matches and no wildcard searching whatsoever. There is no "like" or sql in the search criteria.
- Spaces are not broken into any tables and are not keyed, indexed, in fact space data is never deserialized into object format, it is kept as a serialized stream.

- There are no joins, sql statements, or tables of any sort. There is no ordering as in FIFO, LIFO, or Incrementing counters as there are in databases.
    - **Space Api**
        - Lease l = write(Entry, leaseLength)
        - Entry Object = take(EntryTemplate) + IfExists (non-blocking)
        - Entry Object = read(EntryTemplate) + IfExists (non-blocking)
        - subcription = notify(EntryTemplate, leaseLength)
- All objects in a space have a lease associated with them that when it expires, the entry is reclaimed if its lease is not renewed prior to this timeout. A database does not have this capability.
- Spaces support a notification of an entry written to a space, but doesn't return the actual object that was written to the space, the best that it does it say what type was written.
- There are both transient space implementations as well as persistent space implementations where a restart of the process will retain the entry objects that have yet to expire.

Returning to the claims, it is worth noting that nowhere does Shaughnessy present anything analogous to a Space. Thus, claims 11, 19, 42, and 44 all include features that are not taught or suggested by Shaughnessy.

An additional distinction can be found in the concept that Spaces alert agents, such as the network lurking agent and the network receiving agent, about objects placed in the space for their use. This is mentioned, for example, in claim 16, and supported in the specification at page 8, line 23. Shaughnessy presents no analog to this concept: messages are passed directly between the networks and the central agent server. Nor would the invention be obvious over Shaughnessy, as a Space is more than just a place where data can be temporarily stored. Again, the above description explains the significance of Spaces.

It is also worth noting that in Shaughnessy, there is only one central agent server. In contrast, the invention supports multiple network lurking agents and network receiving agents, all capable of operating independently.

Referring to claim 14, a further distinction between Shaughnessy and the claimed invention can be seen. Claim 14 specifically mentions that an environment setting is stored in the Space. If the Examiner is going to analogize the central agent server to the claimed network lurking agent (a position the Applicant has disputed above), then the central agent

server cannot be the Space. But if the central agent server is not the Space, then settings stored in central agent server are perforce not being stored in a Space.

Referring to claim 16, the Applicant notes that the Examiner rejected the claim as anticipated by Shaughnessy. But nowhere did the Examiner explain where Shaughnessy taught the concept of the Space notifying the network receiving agent about the inquiry being placed in the Space. Accordingly, the Examiner has failed to meet the burden of establishing that the prior art teaches this feature of the claimed invention.

Referring to claim 18, the claim recites the network receiving agent and the network lurking agent as designed to open devices to enable communication. The Examiner has again indicated that the central agent server is capable of implementing all of the features of the claim. But the central agent server of Shaughnessy is described as receiving messages (column 3, lines 25-26), polling networks for available devices (column 3, lines 30-31), and selecting a device to receive the message (column 3, lines 31-32). In other words, the central agent server is a router for messages. Nowhere does Shaughnessy describe the central agent server as opening devices to facilitate communication and indeed, this functionality is beyond the scope of the central agent server. The central agent server is, for example, incapable of selecting the device from which the message originates; by the time the central agent server has received the message, the originating device is selected. This is to be contrasted with the invention, where the network lurking agent is determining if communications can be established; only if communication can be established are devices then selected.

Because Shaughnessy does not teach all of the features in claims 11-14, 16, 18-19, 27-31, and 38-45, claims are allowable under 35 U.S.C. § 102(e) over Shaughnessy. Accordingly, claims 11-14, 16, 18-19, 27-31, and 38-45 are allowable.

For the foregoing reasons, reconsideration and allowance of claims 11-14, 16, 18-19, 27-31, and 38-45 of the application as amended is solicited. **Because the Applicant believes an interview would be helpful, the Examiner is requested to telephone the undersigned at (503) 222-3613 to schedule an interview at his earliest convenience.**

Respectfully submitted,
MARGER JOHNSON & McCOLLOM, P.C.

Ariel S. Rogson
Reg. No. 43,054

MARGER JOHNSON & McCOLLOM, P.C.
1030 SW Morrison Street
Portland, OR 97205
503-222-3613
Customer No. 20575

I hereby certify that this correspondence is being transmitted to the U.S. Patent and Trademark Office via facsimile number 703/872.9306, on October 28, 2004.

Jeanne Bower